ネットワーク上のバックトラックアルゴリズム(計算理論とその応用)

AUTHOR(S):

仙波, 一郎

数理解析研究所講究録
992巻 1997年 215-222

ネットワーク上のバックトラック アルゴリズム

茨城大工学部　仙波一郎（Ichiro Semba）

1．はじめに

ネットワーク上にある多数のワークステーションを同時に使ってバックトラックアルゴリズムを実行するシステムを紹介する。種々の組合せ問題によく応用される部分順列生成アルゴリズム[1,2,3]を例にして、考え方を示す。また、順列生成問題、組合せ生成問題、Nクイーン問題[4,7]、反射的Nクイーン問題[5]、線上三無問題[6,8,9]に対して、このシステムを使った計算機実験の結果を示す。

2．バックトラックアルゴリズム

バックトラックアルゴリズムは、試行錯誤を繰り返しながら解を見落とすことなく、重複することなく探索する一般的な解法手順である。解法手順を木構造で表現すると見通しがよくなり、プログラムが書きやすくなる。集合｛1，2，・・・，n｝上の部分順列生成アルゴリズム［3］の木構造は、図2．1のようになる。

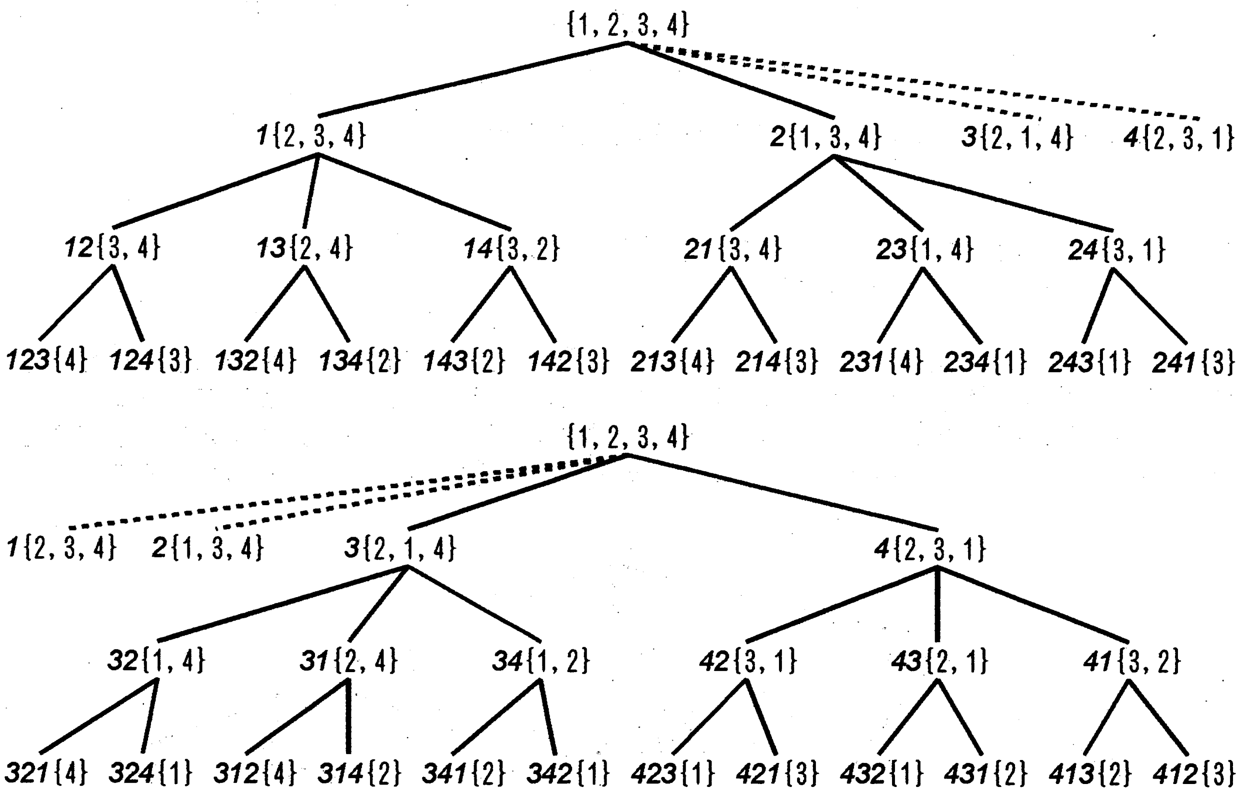

図2．1

このアルゴリズムでは、集合｛1，2，・・・，n｝上の順列の要素をひとつずつ固定していく。すなわち、順列の1番目の要素を固定し、そのもとで、2番目の要素を固定する。この操作をn番目の要素まで続ける。1番目からi番目までの要素が固定された状態を、$a_1 a_2 \cdots a_i \{b_{i+1}, \cdots, b_n\}$ と書くと、この状態からつぎのn－i個の状態が考えられる。

$a_1 a_2 \cdots a_i b_{i+1} \{b_{i+2},\ b_{i+3},\ b_{i+4},\ \cdots,\ b_n\}$
$a_1 a_2 \cdots a_i b_{i+2} \{b_{i+1},\ b_{i+3},\ b_{i+4},\ \cdots,\ b_n\}$
$a_1 a_2 \cdots a_i b_{i+3} \{b_{i+2},\ b_{i+1},\ b_{i+4},\ \cdots,\ b_n\}$
・・・
$a_1 a_2 \cdots a_i b_n\ \{b_{i+2},\ b_{i+3},\ b_{i+4},\ \cdots,\ b_{i+1}\}$

ｉ＋１番目の要素の選び方は、自由であるから多くの方法が考えられる（理論的には、n!・(n-1)!・・・2!・1!通りある）。深さｒの節点を考えるとｎ個の要素からｒ個取り出す部分順列になっており、部分順列の中で要素の並びが昇順のものに注目すると組合せになっている。また、要素の絶対的な位置が保存される特徴がある。

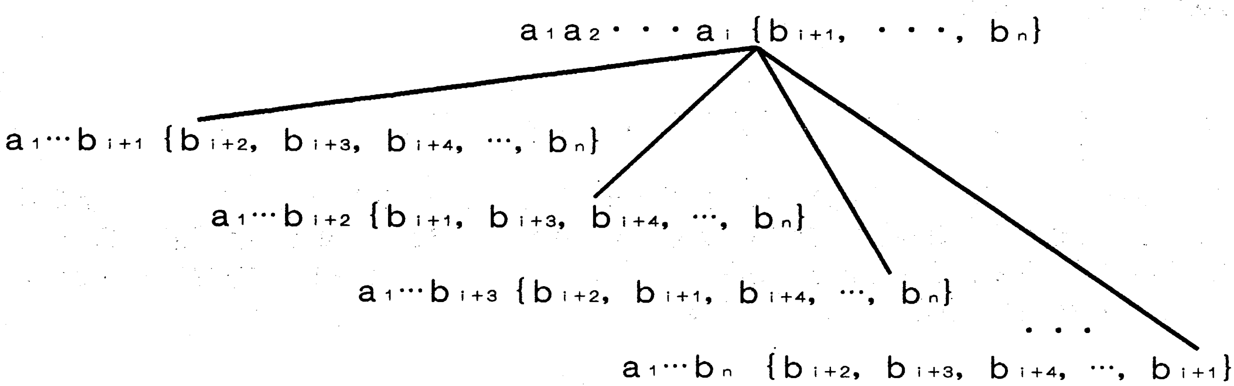

図２．２

部分順列の生成は、木を反時計回りにたどることによって行う。

３．ネットワーク上のバックトラックアルゴリズム

バックトラックアルゴリズムを表現する木構造（図３．１）において、互いに重なり合わない部分木は独立に処理できる。そこで、いくつかの部分木をランダムにまとめグループとし、グループごとにＣＰＵを割り当てれば、同時に多数のＣＰＵを使って、処理することが可能となる。部分木をランダムにまとめるのは、負荷がなるべく偏らないようにするためである。

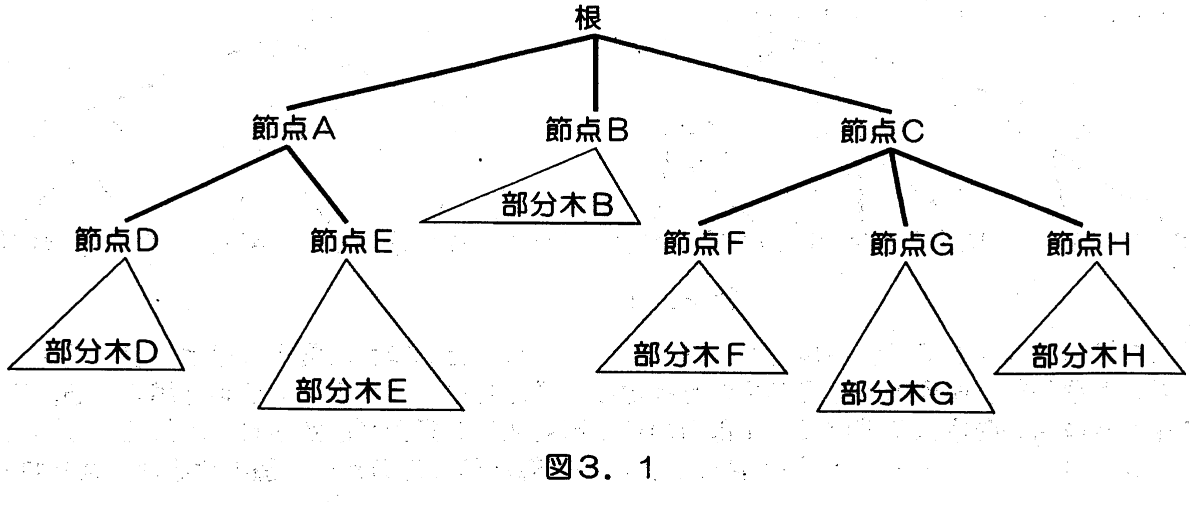

図３．１

４．　仕組み

このシステムが動作するネットワーク環境と設定項目、必要な機能、実行手順、特徴を概説する。

ネットワーク環境

ネットワークに接続され、ＵＮＩＸが動作するコンピュータなら原則として使用できる。
必要な設定は、ローカルホストから簡単に行うことができる。

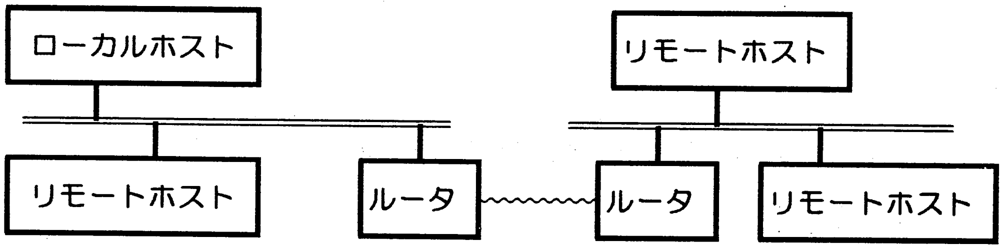

図４．１

● 設定

（１）パスワードなしで、ローカルホスト・リモートホスト間でリモートログインができるように設定する。

● 必要な機能

（１）リモートホストが動作中か停止中か識別できる機能。（ｐｉｎｇコマンド）
（２）リモートホストで動作中のプログラムを強制的に停止させる機能。（ｐｓコマンド、ｋｉｌｌコマンド）
（３）ローカルホストからリモートホストへプログラムとコマンドファイルを転送することができる機能（ｒｃｐコマンド、ｒｓｈコマンド）

● 実行手順

（１）ひとつのＣＰＵで正しく動作するバックトラックプログラムを作成する。
（２）（１）のプログラムを若干手直しして、グループを構成するメインプログラムとグループを実行するサブプログラムをローカルホスト上に作成する。
（３）グループとサブプログラム、それらを実行するシェルプログラムをローカルホストからリモートホストに転送し実行することを繰り返す。
リモートホストは、終了したら結果をローカルホストに転送する。
（４）ローカルホストにすべてのグループの実行結果が返ってきたら終了。

● 特徴

（１）リモートホストにとって、他のユーザと同等である。
（２）割り当てたリモートホストが何らかの理由で停止した場合、ローカルホストは、他のリモートホストに再割当てを行い実行を続けることができる。
すなわち、リモートホストのトラブルに影響を受けずに処理を続行できる。
（３）途中から利用できるようになったリモートホストは、その時点から組み込める。

● 実行例

%init.cmd

```
[ 登録ホストの確認] ：big cis01 hit002
[ 現在のホスト数  ] ： 3
[ 部分問題の作成  ] ：データを入力してください：8 8 1
[  datファイル数  ] ： 8
[tarファイルの作成] ：tarファイル当たりのdatファイル数を指定して下さい：2
[  tarファイル数  ] ： 4
```

%lhost.cmd

```
[分散処理・開始] ：1997年01月22日 (水) 16時32分04秒 JST

tarファイル数は 4 です。
現在利用可能なホスト数は 3 です。

1997年01月22日 (水) 16時32分06秒 JST
tarファイル [1.tar.big] ---> ホスト [big.isemba]
[1] 23312

1997年01月22日 (水) 16時32分09秒 JST
tarファイル [2.tar.cis01] ---> ホスト [cis01.isemba]
[2] 23329

1997年01月22日 (水) 16時32分17秒 JST
tarファイル [3.tar.hit002] ---> ホスト [hit002.isemba]
[3] 23349
*
[1]    Done       rsh big -l isemba csh big.1.cmd 1.tar.big big isemba
[2]    Done       rsh cis01 -l isemba csh cis01.2.cmd 2.tar.cis01 cis01 isemba
[3]    Done       rsh hit002 -l isemba csh hit002.3.cmd 3.tar.hit002 hit002 isemba

1997年01月22日 (水) 16時32分42秒 JST
tarファイル [4.tar.big] ---> ホスト [big.isemba]
[1] 23388

1997年01月22日 (水) 16時32分46秒 JST
tarファイル [4.tar.cis01] ---> ホスト [cis01.isemba]
[2] 23409

総数：40320

[分散処理・終了] ：1997年01月22日 (水) 16時32分55秒 JST

==========================================================================
各リモートホストのプロセスを停止させる処理と確認を行う。
==========================================================================
```

5．計算機実験の結果

ネットワークの環境（使用したコンピュータや混み具合）によって処理時間に変動がある。

部分順列生成問題：集合｛１，２，・・・，N｝上の部分順列をすべて生成する。

N	解の総数	CPU数	処理時間
１２	４７９００１６００	８	３分
１３	６２２７０２０８００	８	２２分
１４	８７１７８２９１２００	８	２時間５３分

表５．１

組合せ生成問題：集合｛１，２，・・・，N｝からR個の要素を選ぶすべての組合せを生成する。

N	R	解の総数	CPU数	処理時間
２６	１３	１０４００６００	８	２分
２８	１４	４０１１６６００	８	３分
３０	１５	１５５１１７５２０	８	９分
３２	１６	６０１０８０３９０	８	３６分
３４	１７	２３３３６０６２２０	８	２時間２９分

表５．２

Nクイーン問題：N×Nのチェス盤上にクイーンが互いにきき筋（縦、横、対角線方向）に入らないように最大何個おけるか、またその置き方は何通りか。

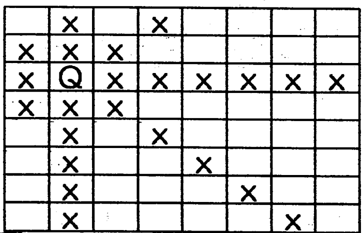

図５．１ クイーンのきき筋

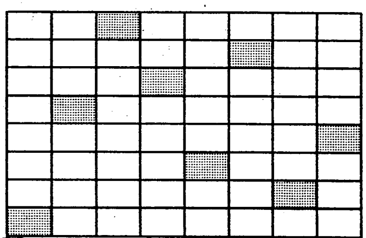

図５．２ ひとつの解

N	解の総数	N	解の総数	N	解の総数
４	２	９	３５２	１４	３６５５９６
５	１０	１０	７２４	１５	２２７９１８４
６	４	１１	２６８０	１６	１４７７２５１２
７	４０	１２	１４２００	１７	９５８１５１０４
８	９２	１３	７３７１２		

表５．３

N	解の総数	CPU数	処理時間
１５	２２７９１８４	８	３分
１６	１４７７２５１２	８	８分
１７	９５８１５１０４	８	５５分
１８	６６６０９０６２４	１３	６時間２３分

表５．４

文献７にはNが１７までの結果が示されている。

反射的Nクイーン問題：Nクイーン問題につぎの条件が付加される。
チェス盤の最上段に１段（反射層と呼ぶ）をつけ加える。
クイーンはそこで反射して動ける。

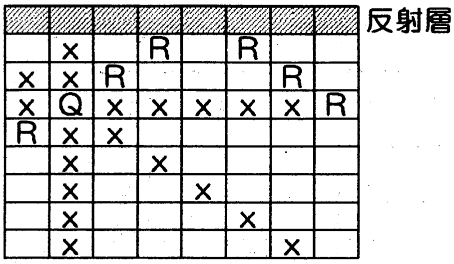

図５．３ クイーンのきき筋

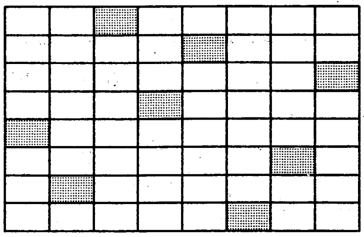

図５．４ ひとつの解

N	解の総数	N	解の総数	N	解の総数
4	2	9	32	14	3178
5	4	10	38	15	16792
6	0	11	140	16	82038
7	2	12	496	17	289566
8	10	13	1186		

表５．５

N	解の総数	CPU数	処理時間
15	16792	8	2分
16	82038	8	4分
17	289566	13	14分
18	1107698	13	1時間13分
19	4804647	8	9時間 9分

表５．６

文献５にはＮが１７までの結果が示されている。

線上三無問題：Ｎ×Ｎのチェス盤上のどの方向にも３個の駒が線上（縦、横、対角線だけでなくあらゆる直線）に並ばないように、最大何個置けるか（最大２Ｎ個は明らか）、またその置き方は何通りか。

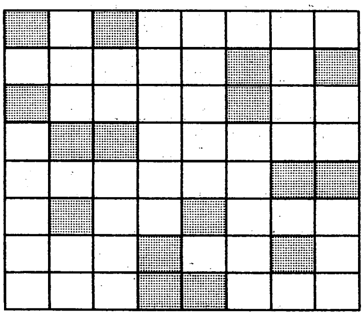

図５．５ ひとつの解

N	解の総数	N	解の総数
2	1	7	132
3	2	8	380
4	11	9	368
5	32	10	1135
6	50		

表５．７

この問題に関して、十分大きなＮに対して、条件を満たす２Ｎ個の駒は存在しないという予想があり未解決である。

●任意の解

文献６にはＮが１１、１２についてまだ解の総数が得られていないことと、Ｎが２６までの解の存在がわかっていると書かれている。

Ｎ	解の総数	ＣＰＵ数	処理時間
１０	１１３５	１８	２５分
１１	１１２０	１８	３時間１４分
１２	４２９５	１９	３２時間５３分

表５．８　すべての解

Ｎ	ＣＰＵ数	処理時間
１３	１８	２７分
１４	１８	２時間２７分
１５	１６	３５時間３７分
１６	１６	５０時間５４分

表５．９　ひとつの解

●点対称解：チェス盤の中心点に対して駒の配置が点対称になる解。

Ｎ	解の総数	Ｎ	解の総数	Ｎ	解の総数	Ｎ	解の総数
２	１	５	０	８	３６	１１	１２０
３	２	６	１８	９	２８	１２	１４４
４	７	７	４０	１０	６７	１３	３３０

表５．１０

Ｎ	解の総数	ＣＰＵ数	処理時間
１４	２７６	１５	１３分
１５	１１３４	１７	５４分
１６	７８４	１８	４時間　７分

表５．１１　すべての点対称解

Ｎ	ＣＰＵ数	処理時間
１６	１７	２分
１７	１７	１０分
１８	１７	１時間３９分

表５．１２　ひとつの点対称解

●対角対称解：チェス盤のひとつの対角線に対して駒の配置が線対称になる解。

Ｎ	解の総数	Ｎ	解の総数	Ｎ	解の総数	Ｎ	解の総数
２	１	５	４	８	１０	１１	１２
３	２	６	４	９	６		
４	３	７	２	１０	９		

表５．１３

Ｎ	解の総数	ＣＰＵ数	処理時間
１２	１０	１５	７分
１３	２０	１５	３２分
１４	１６	１７	２時間　０分

表５．１４　すべての対角対称解

Ｎ	ＣＰＵ数	処理時間
１３	１５	２分
１４	１５	４分
１５	１５	４１分
１６	１５	３時間　３分

表５．１５　ひとつの対角対称解

６．おわりに

バックトラックアルゴリズムをネットワーク上に拡張し、複数のＣＰＵを同時に使っていくつかの新しい結果を得た。

（１）Ｎクイーン問題のＮ＝１８での解の総数。
（２）反射的Ｎクイーン問題のＮ＝１８，１９での解の総数。
（３）線上三無問題でのＮ＝１１，１２での解の総数。

今後の課題として次のようなものが考えられる。

（１）リモートホストが十数台程度のときには、ローカルホスト１台で十分であるが、百台規模になると複数のホストで分担して制御することが必要になる。
（２）線上三無問題について、文献９にはＮの値が２６以下の中で２１，２３，２５において解が知られていないと書かれている。これらの値での具体的な解の発見的な探索と２７以上の値での解の探索が残されている。

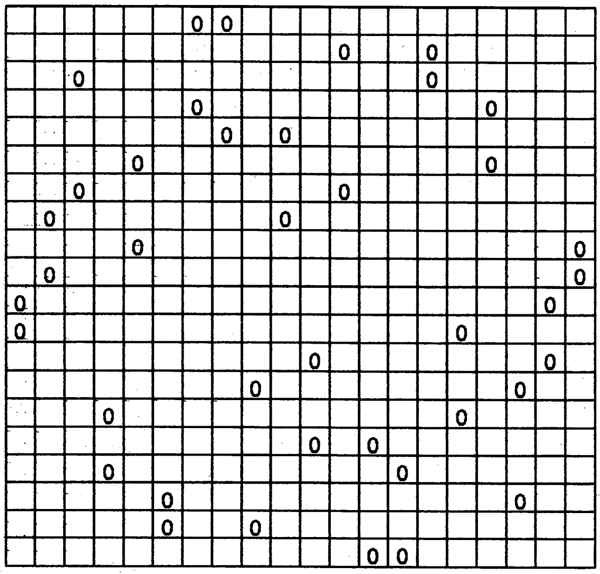

図６．１　Ｎ＝２０の解

参考文献

[１] Sedgewick,R. Permutation Generation Method, Computing Surveys,Vol.9, No.2,1977.（訳 有澤誠）　順列生成の手法、ｂｉｔ（コンピュータ・サイエンス）、vol.10,No.16,1978

[２] Nijenhuis,A. Wilf,H.S. Combinatorial Algotithms 2nd edition, Academic Press 1978.

[３] 仙波一郎,矢島脩三　順列生成アルゴリズムについて、情報基礎理論ワークショップ、7-12, 1992.

[４] R.K.Guy Unsolved Problems in Number Theory, Springer-Verlag, 1981（監訳 一松信）数論における未解決問題集,pp114-116,1982

[５] Gardner,M. Fractal Music, Hypercards and More..., W.H. Freeman and Company, 1991 （訳 一松信）　円周率と詩、丸善、1996.

[６] Gardner,M. Penrose Tiles to Trapdoor Ciphers, W.H. Freeman and Company, 1989,（訳 一松信）　ペンローズ・タイルと数学パズル、丸善、1992.

[７] 仙波一郎、矢島脩三　組合せ問題の論理関数による解法について　計算機構とアルゴリズム、数理解析研究所講究録８３３、204-213,1993

[８] 仙波一郎　チェス盤上の配置問題について、計算量理論、数理解析研究所講究録８７１、248-254, 1994

[９] 芳沢光雄　教養の数学２８講　東京図書、1994